

ABSTRACT

ATTRI, PANKAJ. Detecting Cough in Audio Recordings with Out-of-Distribution (OOD) Data Detection Techniques. (Under the direction of Dr. Edgar Lobaton).

Cough is an important defense mechanism of the respiratory system and is also a symptom of lung diseases, such as asthma. Detecting cough in audio recorded using portable recording devices is a convenient way to track lung health of patients, but the data used in building cough detection models are often clean, contains a limited set of sound categories, and thus the models perform poorly when exposed to a variety of real-world sounds in the audio. The sounds that are not learned by the model are referred to as Out-of-Distribution (OOD) data. In this work, we propose two robust cough detection methods combined with an OOD detection techniques that remove OOD data without sacrificing cough detection performance of the original models. These methods include adding a learning confidence parameter and maximizing entropy loss. Studies conducted in this research also proved that machine learning models can effectively learn and predict cough using data collected by the wearable devices whilst protecting patient privacy by re sampling audio at lower sampling frequencies to inhibit speech recognition. Our experiments show that 1) the OOD system can produce dependable In-Distribution (ID) and OOD results at a sampling rate of 750 Hz and above; 2) the OOD sample detection tends to perform better for larger audio window sizes; 3) the overall accuracy and precision gets better as the proportion of OOD samples increases in the recordings; 4) a higher percentage of OOD data is needed to realize performance gains at lower sampling rates. The incorporation of OOD detection techniques improves cough detection performance by a significant margin and provides a valuable solution to real-world acoustic cough detection problems.

We also explored the hyper parameter space and analyzed effects of using a varying set of Mel Spectrogram generation parameters on model's accuracy in detecting both In Distribution

(ID), i.e., cough and speech, and Out of Distribution (OOD) data. The results prove that there is value in exploring the hyper parameter selection space for all sample rates and that a unique optimal set of hyper parameters should be used for different sample rates before training models. It also proves the fact that an increase in performance of OOD detection further increases the overall ID sample (i.e., cough) classification. This part of the study was limited to only use a sample rate of 750Hz, but the results can be easily extrapolated to other frequencies.

© Copyright 2023 by Pankaj Attri

All Rights Reserved

Detecting Cough in Audio Recordings with Out-of-Distribution (OOD) Data Detection
Techniques

by
Pankaj Attri

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Master Of Science

Computer Science

Raleigh, North Carolina

2023

APPROVED BY:

Dr. Edgar Lobaton
Committee Chair

Dr. Alper Bozkurt

Dr. Min Chi

DEDICATION

I dedicate this thesis to my father who is battling liver cancer. Although the work in this study is not directly related to detecting or curing cancer, I hope it inspires someone to use artificial intelligence to improve health and quality of human lives.

BIOGRAPHY

Pankaj Attri is a Technical Consultant who works full time at SAS and is pursuing his Masters in the Department of Computer Science, NCSU. He completed his bachelor's degree in Chemical Engineering from Indian Institute of Technology, Bombay, in India and moved to the United States in 2015. He is fondly interested of anything related to machine learning and likes to participate in Kaggle competitions. Besides work he likes everything related to Formula One, Tennis, and vacations.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and appreciation to Professor Edgar Lobaton for his support and direction. Besides Dr. Lobaton everyone in the Active Robotics Sensing (ARoS) lab has been very helpful and has played some role in ensuring success of this research.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
Chapter 1: Introduction	1
1.1 Introduction.....	1
1.2 Current and Future state of research	2
1.3 Preliminary work	3
1.4 Contributions	4
1.5 Organization	5
Chapter 2: Related work and approach	7
2.1 Related Work	7
2.2 Proposed Methods	8
Chapter 3: Identifying alternative classifiers and model backbones	10
3.1 Datasets	10
3.2 Setup and Experiments	11
3.3 OOD techniques	14
3.4 Learnings	17
Chapter 4: Experiment with new datasets & implement OOD techniques	18
4.1 Datasets	18
4.2 Setup and Experiments	19
4.2.1 Determine optimal audio duration	20
4.2.2 Comparison of ID data performance on OOD models	22
4.2.3 OOD data performance comparison on OOD models	24
4.2.4 Comparison of model performance on ID data mixed with OOD data	27
4.3 Learnings.....	30
Chapter 5: Compare performance of time frequency functions	31
5.1 Datasets	31
5.2 Setup and Experiments	31
5.2.1 Why were the comparisons performed?	32
5.2.2 Performance comparison	33
5.3 Learnings	34
Chapter 6: Optimization of Mel Spectrograms for cough detection	35
6.1 Datasets	35
6.2 Setup and Experiments	35
6.2.1 Why were the comparisons performed?	36
6.2.2 Model performance on ID data	37
6.2.3 Model performance on OOD data	38
6.2.4 Model performance on In Distribution data with OOD detection	41
6.3 Learnings.....	42

Chapter 7: Conclusions and Future work 43

LIST OF TABLES

Table 1. Dataset Details	10
Table 2. Classifier results comparison	12
Table 3. Model architecture performance	13
Table 4. New Datasets used for Phase II.....	19
Table 5. Number of data points in Train, Validate, Test datasets	20
Table 6. Recall scores comparison.....	30
Table 7. Model performance on In Distribution data for each combination of parameters.	38
Table 8. Model performance on OOD Data for each combination of parameters.	40

LIST OF FIGURES

Figure 1. Current and Future state.	3
Figure 2. Cough detection with OOD	9
Figure 3: FENet architecture.....	11
Figure 4. Confidence score histogram	15
Figure 5. MDS score histogram	17
Figure 6. F1 Score vs Audio interval length	21
Figure 7. AUROC vs Audio interval length	21
Figure 8. TNR@TPR95% vs Audio interval length	22
Figure 9. ID Data Accuracy vs Sampling Frequency	24
Figure 10. ID Data Recall vs Sampling Frequency	24
Figure 11. OOD detection task results comparison	26
Figure 12. Model performance on ID data with varying OOD.....	28
Figure 13. PPSD and FFT spectrogram	33
Figure 14. Performance comparison between PPSD and FFT dataset trained models.....	34
Figure 15. Mel spectrogram images for different parameter combinations.....	37
Figure 16. Model performance with varying OOD proportions	42

Chapter 1: Introduction

1.1 Introduction

The current COVID-19 pandemic is the starkest reminder of the fact that respiratory diseases can severely affect people's health. Diseases like Asthma, COPD, Pneumonia, TB etc. affect or kill millions of children and adults every year [1][2]. In 2019, respiratory diseases comprised three of the top 10 causes of death according to the World Health Organization (WHO), leading to more than 8 million deaths annually. Thus, continuous monitoring of respiratory diseases and early detection, such as that of an Asthma attack, have huge benefits both for patients and physicians alike. But diagnosis of these diseases is often incorrect, and only 25–50% of these patients are known to their doctors [3]. Cough is the most apparent indicator of respiratory diseases and can be detected in audio recording using machine learning models. Audio recordings are usually done by in-home wearable devices [4] but there are some challenges in using data collected by these devices. Real world noises in the recordings largely affect the device and machine learning models' ability to effectively detect cough. A lower performance is expected for a machine learning model if it is applied to data that does not follow the distribution of the target classes used for training. Deep neural network classifiers can give high-confidence predictions to OOD inputs and lead to suboptimal results [5][6]. This is particularly true when switching from a control setting for data acquisition to less structured settings in the real-world. Samples that do not follow the training distribution are not learned appropriately by the models, and they are referred to as Out-of-Distribution (OOD) data. On the other hand, data that the models have seen during training is referred to as In-Distribution (ID) data.

Most of the research related to cough detection focuses on specific sound classification problems and assumes the data are clean but low data quality caused by ambient environmental

noise affects model performance. In this study we introduce an OOD detection techniques to tackle this issue. These techniques are embedded in machine learning models such that the model can classify cough after discarding any OOD samples.

The models developed in this study are primarily image classification models and so we transform the audio recordings into 2D images, images that can be used to train the models. We used two time-frequency analysis methods to convert audio recordings into 2D images: Fourier transformation (FFT) and Power Spectral Density (PSD). The images generated using these techniques are called Spectrograms/Mel-Spectrograms. We used PyTorch packages for Short-term Fourier Transformations (St-FFT) and collaborated with SAS to get Short-term Power Spectral Density (St-PPSD) transformations of audio signals. Majority of the study was carried using St-FFT but we also compared performance of St-FFT based datasets to St-PPSD based datasets to find out if one performs better than the other.

Lastly, we conducted additional experiments to analyze the effect of changing the Mel Spectrogram generation parameters on model performance. The models trained in our study were initially all trained on datasets generated using a fixed set of parameters. In this part we created multiple datasets using a variable set of parameters and studied their impact on performance. The results indicate that there is tremendous value in exploring the parameter space to create even more accurate models. This part of the study focused only on the 750Hz resampled audio signals, but the results can be easily extrapolated to other sample rates.

1.2 Current and Future state of research

Fig. 1 shows current and future state of the research work.

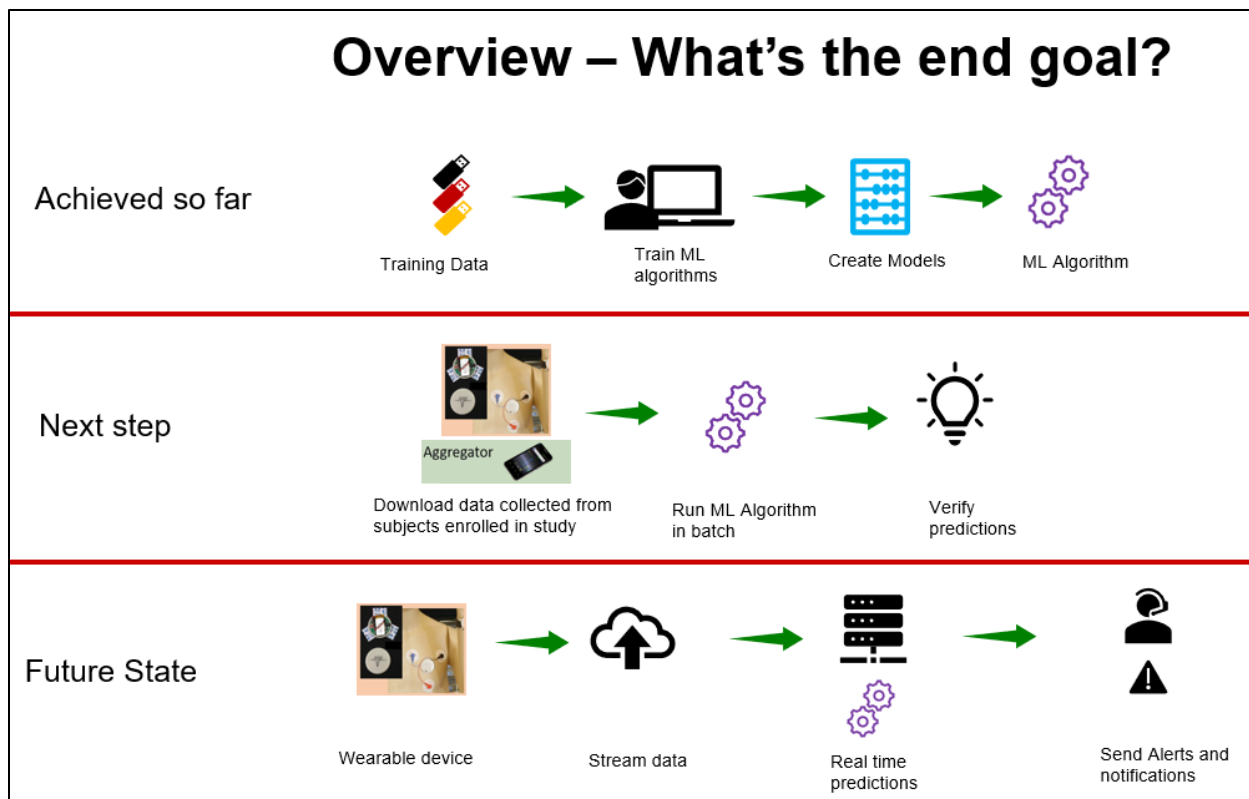


Figure 1. Current and Future state.

The stream at the top is what we have achieved so far where we have used publicly available datasets to create neural network models that have OOD techniques to classify cough in audio recordings. As a next step we will test these models on recordings done by actual human beings who are enrolled as part of the larger study. In the future state, we plan to use our models to do real time predictions on audio recordings so that physicians and/or patient’s family members can be notified in real time for cases like Asthma attacks where notifying physicians of an onset of an Asthma attack can potentially save the patient’s life.

1.3 Preliminary work

ARoS (Active Robotics Sensing Lab) lab created neural network models trained on ESC 50[9], FSDKaggle2018[10], LibriSpeech[11] datasets and concluded [12] that a down sample rate

of 750 Hz is enough to convincingly obfuscate speech for privacy purposes maintaining a specificity, sensitivity, and accuracy of 0.92 + for cough classification. The models were CNN based neural network based on FENet architecture (pretrained on Google Audio dataset). Input data had audio segments of 1.5 sec interval. The work did not include any OOD detection techniques and focused only on classifying cough and speech samples in the datasets.

1.4 Contributions

The main contributions of this study include the following:

- We used following publicly available datasets: Musan [37], Coughvid [35], and FluSense [7]. Datasets were created for several frequencies and audio length intervals to train models that detect both In-Distribution and Out-of-Distribution data.
- The OOD detection task was embedded in the same model that performs ID data classification. We implemented two different OOD techniques to compare their performance against baseline models that do not have any OOD detection capabilities.
- We experimented with multiple sampling frequencies to find out the lowest sampling frequency that can be used to create models to convincingly detect cough while inhibiting speech recognition to ensure no patient privacy information is leaked. We showed that the cough detection with OOD detection can produce reliable results at above 750 Hz sampling rate at 1.5 - 10 seconds window sizes.
- We demonstrated that the performance of baseline models starts to deteriorate as the OOD data proportion increases in the data passed in the models for testing. The gains become more prominent at higher sample rates and for higher proportions of OOD data.
- We compared performance of regular spectrograms generated using St-FFT transformations with St-PPSD based spectrograms and concluded that neither

outperforms the other in all metrics. The caveat here is that we tested only with 3 sampling frequencies. Generating PPSD based datasets takes a lot of time and that restricted us to do a fair comparison between the two. More work in this area is planned for next phase.

- We analyzed the performance impact of varying Mel Spectrogram generation parameters on model performance and concluded that there is tremendous value in exploring the hyper parameter space to find an optimal parameter combination for a sampling frequency.

1.5 Organization

Research work in this study is split in four different phases. Each phase is detailed in a separate chapter. For each phase, experiment, dataset, and result details are provided. Individual chapter details are below:

- Chapter 2: There are two parts in this chapter. The first part provides details on the work research community has already done for cough detection including OOD data task. The second part provides details about the proposed methods that we implemented in the study.
- Chapter 3: Provides details of Phase I of the work, which was focused on improving preliminary work, i.e., cough detection pipeline and exploring OOD detection techniques.
- Chapter 4: Phase II of the work revolves around experiment with more robust datasets, different model architectures, and implementation of OOD techniques.

- Chapter 5: This chapter has details on Phase III of the work where we collaborated with SAS to find out whether SAS St-PPSD transformed spectrogram images perform better than St-FFT based Spectrogram images.
- Chapter 6: Phase IV of the work. Focused on conducting experiments to analyze impact of Mel Spectrogram generation parameters on model performance.
- Chapter 7: Conclusions and future work.

Chapter 2: Related work and approach

2.1 Related Work

Cough detection has been studied widely using traditional statistical techniques [13][14], but researchers have also started to implement machine learning algorithms to learn audio signal features that can be used to detect cough [20] [21]. Some of these works use Short-time Fourier transform (STFT), Mel-frequency cepstral coefficients (MFCC) and Mel-scale filter banks (MFB) to create input datasets. The actual models use a variety of classifiers including logistic regression, feed-forward artificial neural network, support vector machine, and random forest. Other works by Monge-Alvarez et al. [22] propose using high-level data representation steps to enhance cough detection performance and Lee et al. [23] proposed adding data augmentation processes. [24] also used an ensemble of classifiers to work around hardware device issues and the cross-device discrepancies. In the preliminary work conducted by the ARoS lab, Mahmoud et al. [12] demonstrated the effectiveness of a logistic regression classifier with Mel-spectrogram input applied to cough and speech audio using various sampling frequencies and window sizes. This work also uses Mel-spectrograms of audio signals as inputs to machine learning models based on Resnet, VGG, and EfficientNet based architectures with custom OOD detection techniques to improve overall cough classification performance.

There have been several studies conducted to handle the problem of OOD data in the world of computer vision [25] – [34]. Some of these works did not change the underlying model architectures and instead rely on using softmax function to separate OOD data [25][26], while others add an additional output indicating the confidence of the results to identify OOD inputs [27]-[29]. In this work we implemented two OOD detection techniques. In the first approach, we add a learning confidence output [27] to the ID cough prediction branch of the model and in the

second the SoftMax loss is replaced by a custom classification layer and a custom loss function called IsoMaxPlus loss [19], which adheres with the principles of maximum entropy. The classification layer learns a prototype for each class and uses distance as a measure to separate OOD and ID data.

2.2 Proposed Methods

Machine learning models we created in this study tackle cough detection and out-of-distribution detection problems simultaneously. The workflow of the proposed pipeline is shown in Fig 2. In our proposed pipeline, we use Mel-spectrogram [40] as audio inputs to a Convolutional Neural Network (CNN) based models to extract features for cough and out-of-distribution detection tasks. The output is first analyzed to determine whether it meets a criterion to be classified as OOD, and if it does then it is discarded before the model classifies the rest as cough or non-cough. We experimented with different CNN based models as backbones along with two classifiers that have OOD detection capabilities. For CNNs, we test the Frequency Extraction Network (FENet) in [18], Residual Network [17], EfficientNet [31] and VGG [36] based models. We test our pipeline on two different datasets. To find the best CNN model backbone, we used a dataset from our earlier pipeline [12] and to find out the best model (for cough classification and OOD detection) we used a different set of datasets. We also investigated how different sampling rates and window sizes of the audio signals affect the results.

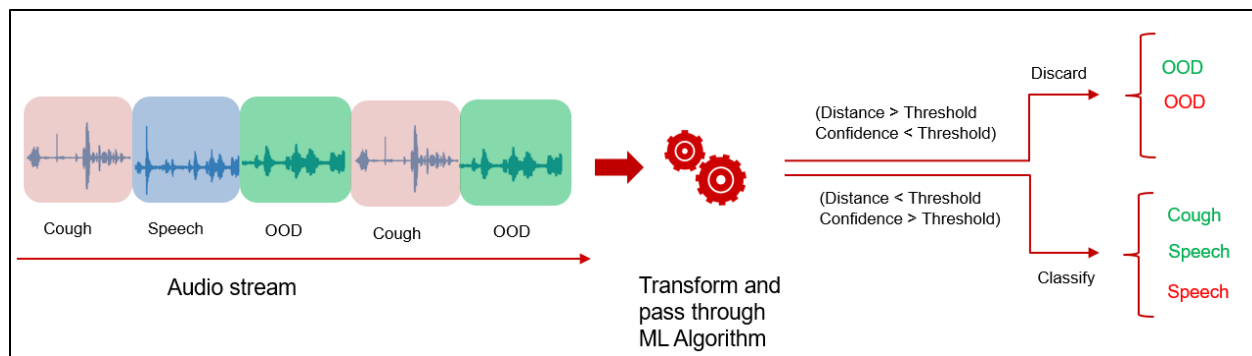


Figure 2. Cough detection with OOD

Chapter 3: Identifying alternative classifiers and model backbones

In this chapter we explain work done in Phase I, which was focused on improving models created in the preliminary work. We identified two areas to explore for improvements. The first one focused on improving the classifier used in the FENet based models and the second one focused on exploring alternative model backbones like ResNet and VGG based architectures to compare performance of FENet architecture.

3.1 Datasets

Table 1 lists the source data used for generating datasets for this phase. ESC 50[9], FSDKaggle2018 [10], LibriSpeech [11] are publicly available datasets. All audio files are converted and saved in WAV format. The length of audio files in the datasets varies from 5 to 30 seconds. For the first phase we extracted 1.5 second audio intervals from these files. Note that the optimal length of audios was investigated in [12]. The audios were manually annotated for cough signals by student assistants. The cough dataset has 8,046 data points extracted from both the ESC-50 and FSDKaggle2018 datasets. 11,372 speech data points were extracted from the LibriSpeech dataset. No changes were made to the process of extracting data as used in [12]. The dataset was split into training (80%), validation (10%), and testing (10%) subsets. The audio clips once generated were transformed to Mel-Spectrogram 2D images using PyTorch packages.

Table 1. Dataset Details

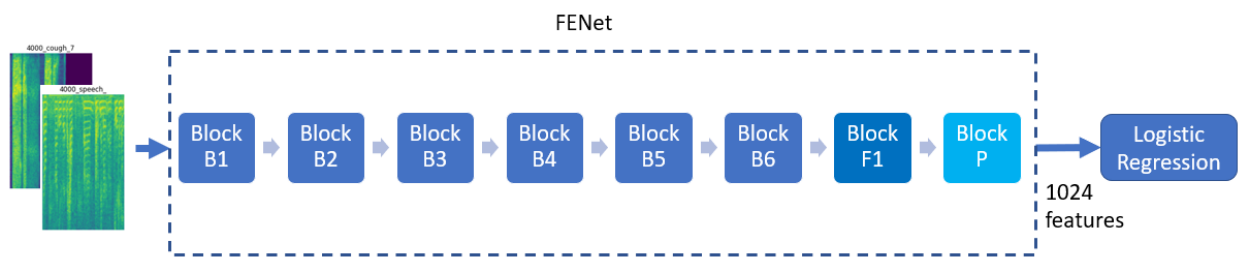
Data Type	Audio Type	Dataset
In-Distribution (ID)	Cough	ESC-50
		FSDKAGGLE 2018
	Speech	LibriSpeech

Table 1. (continued).

Out-of-Distribution (OOD)	Respiratory	RESP
---------------------------	-------------	------

3.2 Setup and Experiments

As stated above the first part focused on experimenting with classifiers to find out if the existing logistic regression classifier used in the FENet model can be improved. All experiments were run on Colab notebooks with GPU accelerators and using PyTorch packages. The FENet architecture is shown in Fig. 3. As explained in [12] blocks B1 through B5 each consist of two cascaded copies of a convolutional layer followed by batch normalization and then a ReLU activation, with 16, 32, 64, 128, and 256 feature maps respectively. Block B6 consists of a convolutional layer with 512 feature maps followed by batch normalization, a ReLU activation, and then max pooling. Block F1 consists of a convolutional layer with 1024 feature maps followed by batch normalization and a ReLU activation and block P is a global averaging pooling layer that outputs a 1024 feature map for each sample passed through the network.

**Figure 3:** FENet architecture

For experiments with new classifiers, we tested two decision trees-based classifiers: XGBoost and Random Forest. Input to these classifiers were the 1024 high level features obtained from block P of the FENet model as shown in Fig. 3. The experiments were performed only for

one sampling frequency of 16,000 HZ for 1.5s audio recordings. Results obtained with the three classifiers are noted in Table 2. Note that the results are model's performance in recognizing cough and speech data with cough being the True Positive (TP) class. The results from the testing proved that considering all metrics the logistic regression classifier outperformed other classifiers.

Table 2. Classifier results comparison

FENet with Logistic Regression		XGBoost (default params)		Random Forest (max features = 32, n_e = 50)	
Discriminator Stats		Discriminator Stats		Discriminator Stats	
Sensitivity/Recall	0.955	Sensitivity/Recall	0.968	Sensitivity/Recall	0.923
Specificity	0.973	Specificity	0.946	Specificity	0.971
Accuracy	0.966	Accuracy	0.955	Accuracy	0.951
Matthew's correlation coefficient	0.929	Matthew's correlation coefficient	0.909	Matthew's correlation coefficient	0.899
Precision/PPV	0.962	Precision/PPV	0.927	Precision/PPV	0.957
NPV	0.968	NPV	0.976	NPV	0.947

The other set of experiments that we conducted tested different model backbones as alternatives to the FENet architecture. Architectures that we tested were ResNet and VGG. For this part we used the 16,000 Hz sampling frequency with 1.5 sec audio intervals. The results are shown in Table 3.

Table 3. Model architecture performance

Framework	FENet	ResNet18	VGG16
Epochs	1	1	1
Sample Rate	16000	16000	16000
Pretrained	Yes	Yes	Yes
Classifier	Log Regression	Log Regression	Log Regression
Discriminator Stats			
Train loss	0.005	0.0001	0.001
Valid loss	0.004	0	0.004
Test loss	0.090	0.001	0.084
Sensitivity/Recall	0.955	1	0.950
Specificity	0.973	1	0.998
Accuracy	0.966	1	0.978
Balanced accuracy	0.964	1	0.974
Matthew's correlation coefficient	0.929	1	0.955
Precision/PPV	0.962	1	0.997
NPV	0.968	1	0.966

As is clear from the results in Table 3 both ResNet and VGG architecture based model outperformed FENet models. This result helped us identify that we needed to move away from FENet architecture and use ResNet based models for the remaining study.

3.3 OOD techniques

In addition to the experiments to find better classifiers and model architectures Phase I also focused on researching OOD data detection techniques. Two techniques were shortlisted. The first is adapted from the learning confidence out-of-distribution detection model [27], which estimates learning confidence for neural networks and produces intuitively interpretable outputs. It generates a confidence value for each predicted output in the range of 0 to 1. A confidence value close to 1 indicates that the data is more likely to be an in-distribution data and a score close to 0 indicates that the data is more likely an OOD data. Learning confidence is estimated by adding a confidence estimation branch along with the original class prediction branch after the second to last layer of the original network. To test this technique, we used a ResNet-18 model and replaced the last layer with two separate layers. One controlling the prediction task and the other the confidence estimation task. The prediction layer is a linear layer with two outputs followed by SoftMax function that generates output probabilities p and the confidence layer is again a linear layer with single output followed by a Sigmoid function that generates a confidence score c . This process can be represented as:

$$p, c = f_{\theta}(x), \quad p_i, c \in [0,1], \quad \sum_{i=1}^2 p_i = 1 \quad \text{Eq.1}$$

where x is the Mel spectrogram input and θ represents the parameters for the neural network. After adding a dependency on the confidence score c , the modified predicted probability becomes:

$$p' = c \cdot p + (1 - c) \cdot y \quad \text{Eq. 2}$$

where y is the true class. A value of c close to 1 indicates that the modified predicted probability is close to the predicted probability, in other words the results are more convincing,

and the input data is ID. On the other hand, a value close to 0 will push the modified predicted probability towards the ground truth, which highlights that the predictions are not accurate, and the input data is OOD. The binary cross entropy loss using predicted probability becomes:

$$L_t = - \sum_{i=1}^2 \log(p'_i) \cdot y_i \quad \text{Eq. 3}$$

To ensure that the model does not try to set the value of c close to 0 in Eq. 1 we add a penalty to the confidence score c .

$$L_c = -\log(c) \quad \text{Eq. 4}$$

The total loss function then becomes:

$$L = L_t + \lambda \cdot L_c \quad \text{Eq. 5}$$

where λ is a hyper-parameter. Fig. 4 shows a typical confidence distribution histogram for ID and OOD samples. It is easy to notice that the majority of OOD samples have a confidence score that is closer to 0 and ID samples are more closer towards a confidence score of 1.

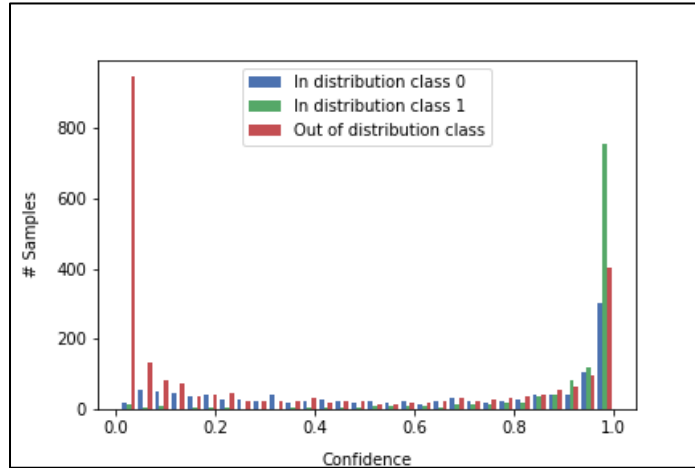


Figure 4. Confidence score histogram

The second OOD detection technique [19] employs a custom classification layer and a custom loss function as shown in equation 6 below.

$$L_{IsoMax} = - \sum_i \log \left(\frac{\exp(-d(f_\theta(x), z_\phi^i))}{\sum_j \exp(-d(f_\theta(x), z_\phi^j))} \right) \cdot y_i \quad \text{Eq. 6}$$

where $f_\theta(x)$ denotes the embedded high-level features and z_ϕ^i denotes a learnable prototype of class j . These prototypes are learned during training process by minimizing loss function. The function $d(., .)$ is equal to the nonsquared Euclidean distance between sample features and class prototypes, and y stands for the correct class label. This loss function is a drop in replacement for the SoftMax loss function because it follows the same normalization and logits structure. The negative notation for the distance measure indicates an inverse relation with the logit because a smaller distance value to the class prototype will place a sample close to the correct class prototype. An extension on the IsoMax loss, called IsoMaxPlus loss is also proposed by [19] in which $f_\theta(x)$ and z_ϕ^i are replaced with their normalized versions making the loss function to be:

$$L_{IsoMaxPlus} = - \sum_i \log \left(\frac{\exp(-E_s \beta \|\hat{f}_\theta(x) - \hat{z}_\phi^i\|)}{\sum_j \exp(-E_s \beta \|\hat{f}_\theta(x) - \hat{z}_\phi^j\|)} \right) \cdot y_i \quad \text{Eq. 7}$$

Where E_s is an entropic scale and β is a learnable parameter. The distance measure, called Minimum distance Score (MDS) is then used to calculate a distance threshold to distinguish ID from OOD data. The MDS is give be:

$$MDS = \min_j (\|\hat{f}_\theta(x) - \hat{z}_\phi^j\|) \quad \text{Eq. 8}$$

Fig. 5 shows an example of the distance distribution histogram for ID and OOD samples. You will notice that the ID-cough, and ID-cough samples are much closer to the origin than the OOD samples.

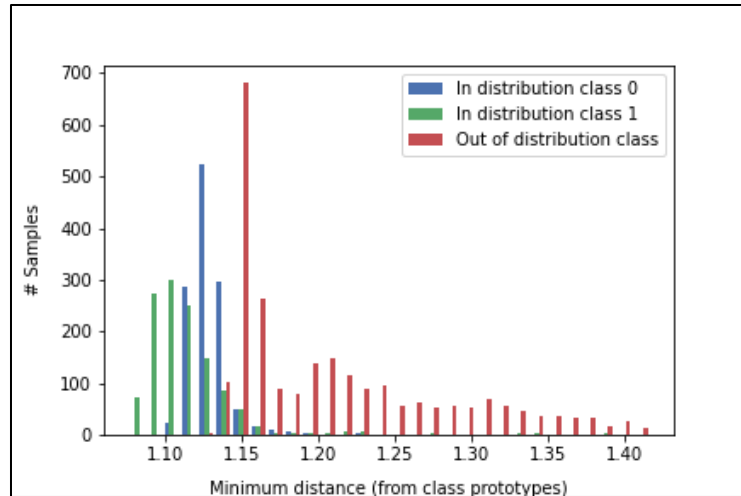


Figure 5. MDS score histogram

3.4 Learnings

Experiments conducted in this phase proved that we need to use other model architectures than the FENet architecture used in the preliminary study. The other learning was that we need to use more robust datasets because we got a perfect score of 1 on all metrics with the ResNet-18 model, which highlights that the dataset did not have enough variation and that the models were overfitting.

Chapter 4: Experiment with new datasets & implement OOD techniques

Phase II of the work concentrated on experimenting with new datasets and implementing OOD techniques on the new datasets. Sections below provide details on the datasets used and the results obtained on cough classification with OOD data detection techniques.

4.1 Datasets

We used four different data sources for generating datasets for this phase. Coughvid dataset [35], FluSense dataset [7], the Musan dataset [37], and the LibriSpeech dataset [11]. Cough and speech samples were generated from Coughvid & FluSense, and from Musan dataset & LibreSpeech datasets respectively. These two classes combined formed our in-distribution dataset. Coughvid dataset was recently collected with over 25,000 crowd sourced cough recordings and contributes as one of the largest expert-labeled cough datasets in existence and data in FluSense was collected by a low-cost microphone sensor attached to a Raspberry Pi. The Musan dataset contains a corpus of music, speech, and noise. Music and noise are used to generate out-of-distribution datasets. Audio clips are generated using sliding windows. The training, validation, and test datasets follow a 75%, 15%, and 10% splits respectively. OOD detection techniques applied to the models do not need OOD data for training and validation purposes so the only dataset that contains OOD data is the test dataset. The proportion of OOD data as compared to ID data is varied from 0% to 100% for some experiments and 0-50% for others, where at 100% the proportion is 2:1 where the number of OOD samples equals number of cough samples plus number of speech samples. We created multiple datasets by resampling audio clips at various sampling frequencies {250, 400, 500, 750, 1k, 2k, 4k, 6k, 8k, 10k, 12k, 14k, 16k} Hz and by using various audio intervals of {1.5, 2, 3, 4, 5, and 10} seconds. The audio clips once generated were transformed to Mel-

Spectrogram 2D images using PyTorch packages. Converting audio clips to Mel-Spectrograms was necessary to use image classification models for cough and OOD detection tasks.

Table 4. New Datasets used for Phase II

Data Type	Audio Type	Dataset
In-Distribution (ID)	Cough	Coughvid
		FluSense
	Speech	Musan (Speech) LibriSpeech
Out-of-Distribution (OOD)	Music & other random noises	Musan (Non Speech)

We experimented with multiple sampling frequencies because an additional intent of the study was to find the minimum sampling frequency such that the cough and OOD detection tasks perform well while inhibiting speech recognition to prevent any loss of privacy information in audio recordings.

4.2 Setup and Experiments

As noted above we experimented with multiple sampling frequencies and audio intervals, but it was not productive and would take enormous amounts of time to test all possible combinations. To work around this problem, we chose to experiment with a subset of sampling frequencies $\in \{400, 750, 4000, 16000\}$ to find out the optimal audio interval length, $\tau, \in \{1.5, 2,$

3, 4, 5, 10} seconds that we should work for the rest of the study. The total number of samples in train, validate, and test datasets for different τ are shown in Table 5.

Table 5. Number of data points in Train, Validate, Test datasets

	$\tau(s)$	1.5	2	3	4	5	10
Training dataset	Cough	9522	8204	9918	9384	7074	6282
	Speech	9522	8204	9918	9384	7074	6282
Validation dataset	Cough	1602	1387	1669	1556	1124	1095
	Speech	1602	1387	1669	1556	1124	1095
Test dataset	Cough	1086	931	1057	961	802	748
	Speech	1086	931	1057	961	802	748
	OOD	2172	1862	2114	1922	1604	1496

For the neural network models, we chose to work with a deeper ResNet-50 architecture with OOD techniques as described in earlier chapters. Both models were based on ResNet-50 architecture. The models are referred to as Entropy Based and Confidence Based, where the Entropy Based model has IsoMaxPlus loss based OOD detection technique and the Confidence Based model has the confidence score OOD detection technique applied. Each model was trained for 6 epochs using a fixed learning rate of $1e-04$ with Adam optimizer.

4.2.1 Determine optimal audio duration

Fig. 6, 7, and 8 show results of experiments to find the optimal audio duration or length. The experiments were conducted with both OOD models. Fig. 6 shows the F1 Score value obtained when testing on ID data, with cough being the True Positive class. It's clear from Fig. 6 that the F1 Score value improves generally for all models as the audio interval length is increased for all

sampling frequencies but audios with 4 second or higher seem to give better performance than smaller length audio clips.

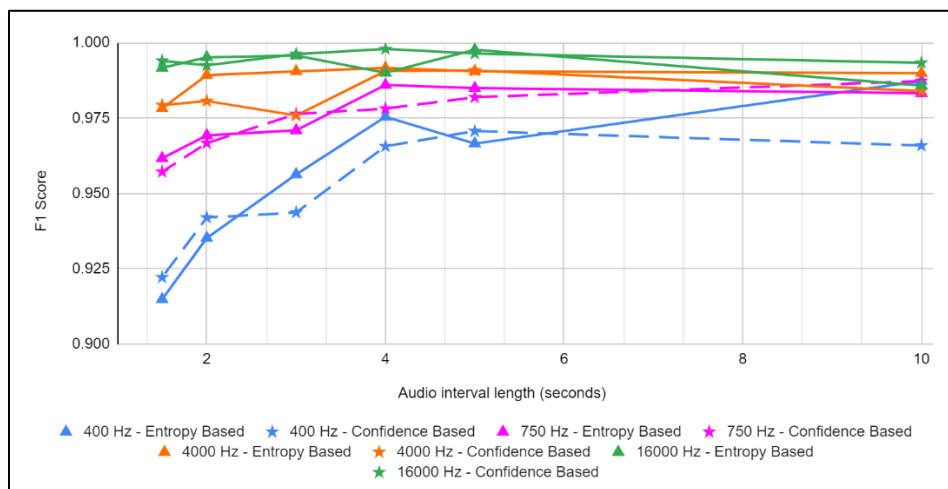


Figure 6. F1 Score vs Audio interval length

Fig. 7 shows the AUROC values where the TPR and FPR values calculated using In-Distribution data as the True Positive class and OOD data as True Negative. This data is obtained when a mixture of ID and OOD samples are passed through the models and a threshold MDS or confidence score value is determined that gives the maximum AUROC value.

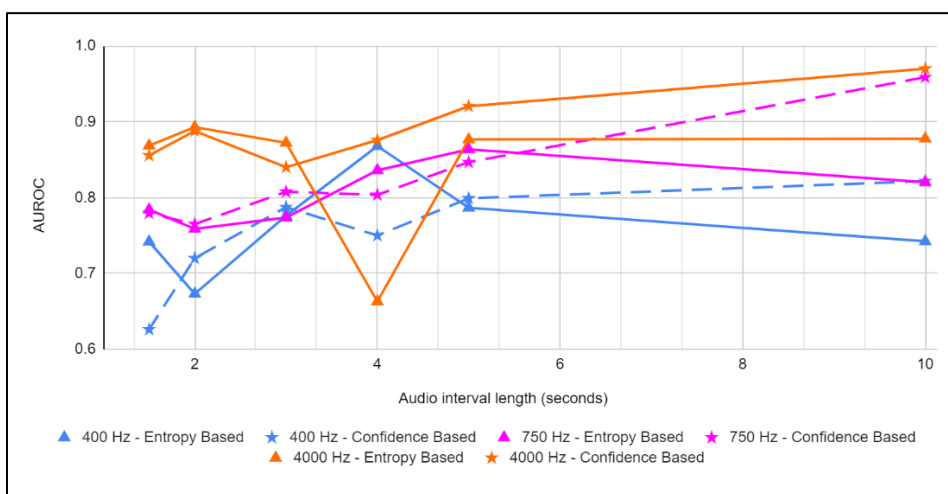


Figure 7. AUROC vs Audio interval length

Fig. 8 plots the TNR@TPR 95% scores for different audio interval lengths for three sampling frequencies. True Negative Rate ($TNR = TN / (FP + TN)$) at 95% True Positive Rate ($TPR = TP / (TP + FN)$) is interpreted as the probability that an OOD sample is correctly classified as OOD when the true positive rate (TPR) is as high as 95%. Both Figure 7 and 8 follow the same trend which suggests using a 4, 5, or 10 second audio interval gives the best performance. For the rest of the study, we decided to only use 5 second audio interval lengths.

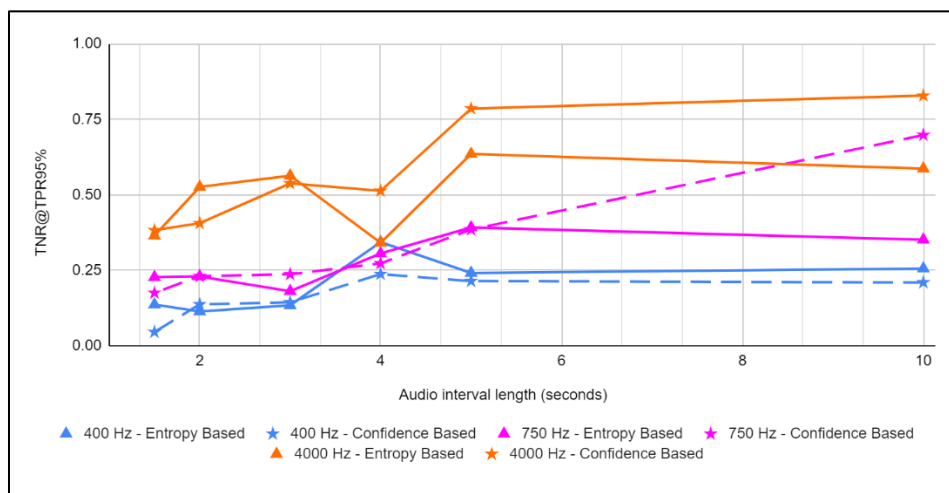


Figure 8. TNR@TPR95% vs Audio interval length

4.2.2 Comparison of ID data performance on OOD models

The next step in the work was to find out whether adding OOD techniques in machine learning models affects model's performance on ID data. For this comparison we created a baseline model which is based on ResNet-50 architecture but with a simple linear layer in the end (followed by SoftMax) for binary classification purposes. The performance of the baseline model was compared with two other OOD models explained above, which are ResNet-50 Entropy based OOD technique model and ResNet-50 Confidence based OOD technique model. The comparison was done by creating models with datasets created at various sampling frequencies $\in \{250, 400, 500,$

750, 1k, 2k, 4k, 6k, 8k, 10k, 12k, 14k, 16k} Hz. 4 models, with different starting seeds, were trained for each sampling rate for 6 epochs at a fixed learning rate of $1e-04$. The results reported for this experiment are simple means of the scores obtained on the test dataset. Note that while training and validating the models, only ID data was used and the model that gave best scores on the ID validation dataset was chosen as the best model to run on the test dataset. Also note that the test dataset has both ID and OOD data in them. Fig. 9 shows performance comparison of the three models on ID data for various sampling frequencies. The model performance is not too different at higher sampling rates, although the confidence-based model (in yellow) gives somewhat lower performance than the other two. Fig. 10 shows the Recall values for the three models. There is more variation in this measure when compared to the Accuracy graph. There is not a consistent trend where one performs better than other over all sampling frequencies. It does seem to appear that at higher sampling frequencies the difference in the performance is negligible. The results from this experiment highlight that including OOD detection techniques in baseline models should not impact model's performance on ID data, and in fact the performance should increase as models encounter OOD data. Models with OOD detection techniques should be able to discard OOD samples before making an ID data class prediction, thus making fewer mistakes compared to the model that does not have an ability to discard OOD data.

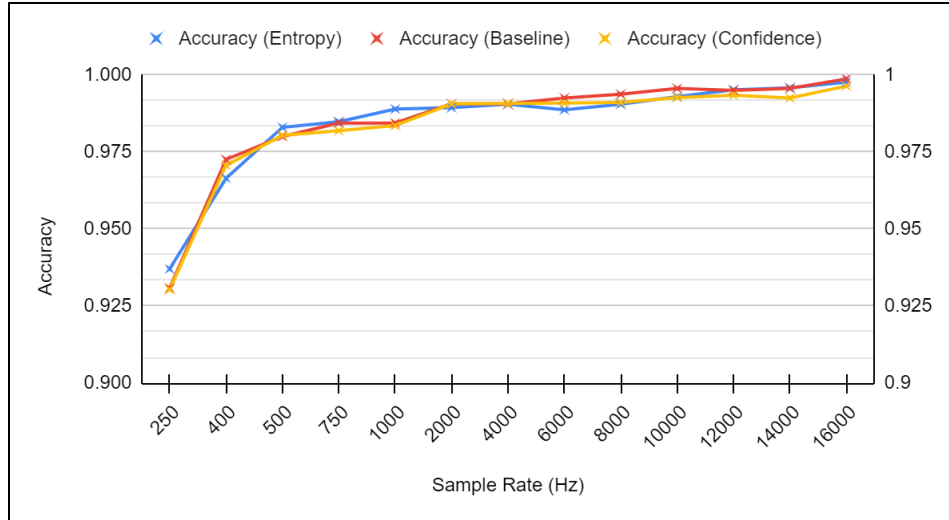


Figure 9. ID Data Accuracy vs Sampling Frequency

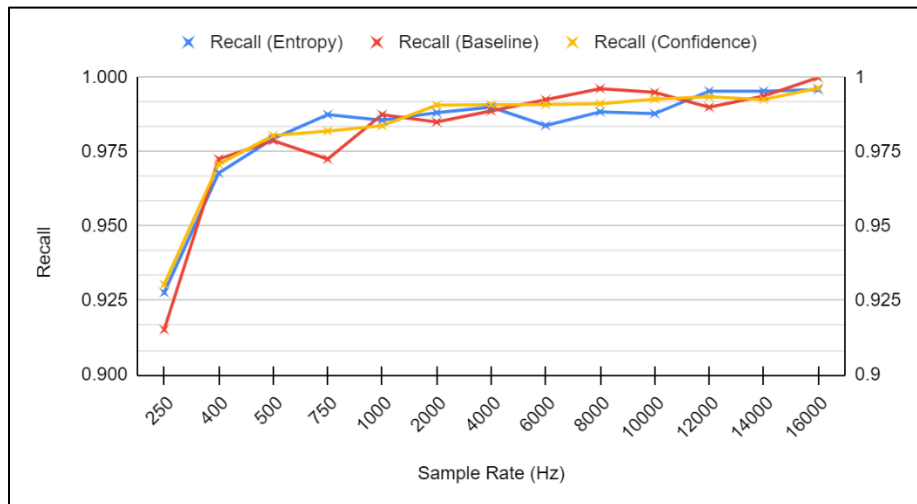


Figure 10. ID Data Recall vs Sampling Frequency

4.2.3 OOD data performance comparison on OOD models

The OOD detection performance of the models is tested next. Fig. 11 shows the results of mean and standard deviation of 4 confidence-based and entropy-based models. The final model for each run is selected by looking at performance of ID validation dataset. The performance measure tracked is overall classification accuracy. For the OOD detection task OOD samples are

detected by using a distance threshold calculated using the AUROC curves obtained with test dataset. Similarly, for confidence-based models that threshold is determined using the confidence score. The results in the Fig. 12 for detection error, F1, precision, and recall use these thresholds. Note that the FPR@95% TPR is nothing but $1 - \text{TNR@95\% TPR}$, as reported before. The Detection error on the other hand is defined in Eq 9, where $\delta \in [0, 1]$ are all possible thresholds. $f(x)$ is the confidence score for an input sample x and P_{in} and P_{out} are the classification probability of ID and OOD examples respectively, and they are equally weighted with 0.5:

$$\min_{\delta} \{ 0.5P_{in}(f(x) \leq \delta) + 0.5P_{out}(f(x) > \delta) \} \quad \text{Eq. 9}$$

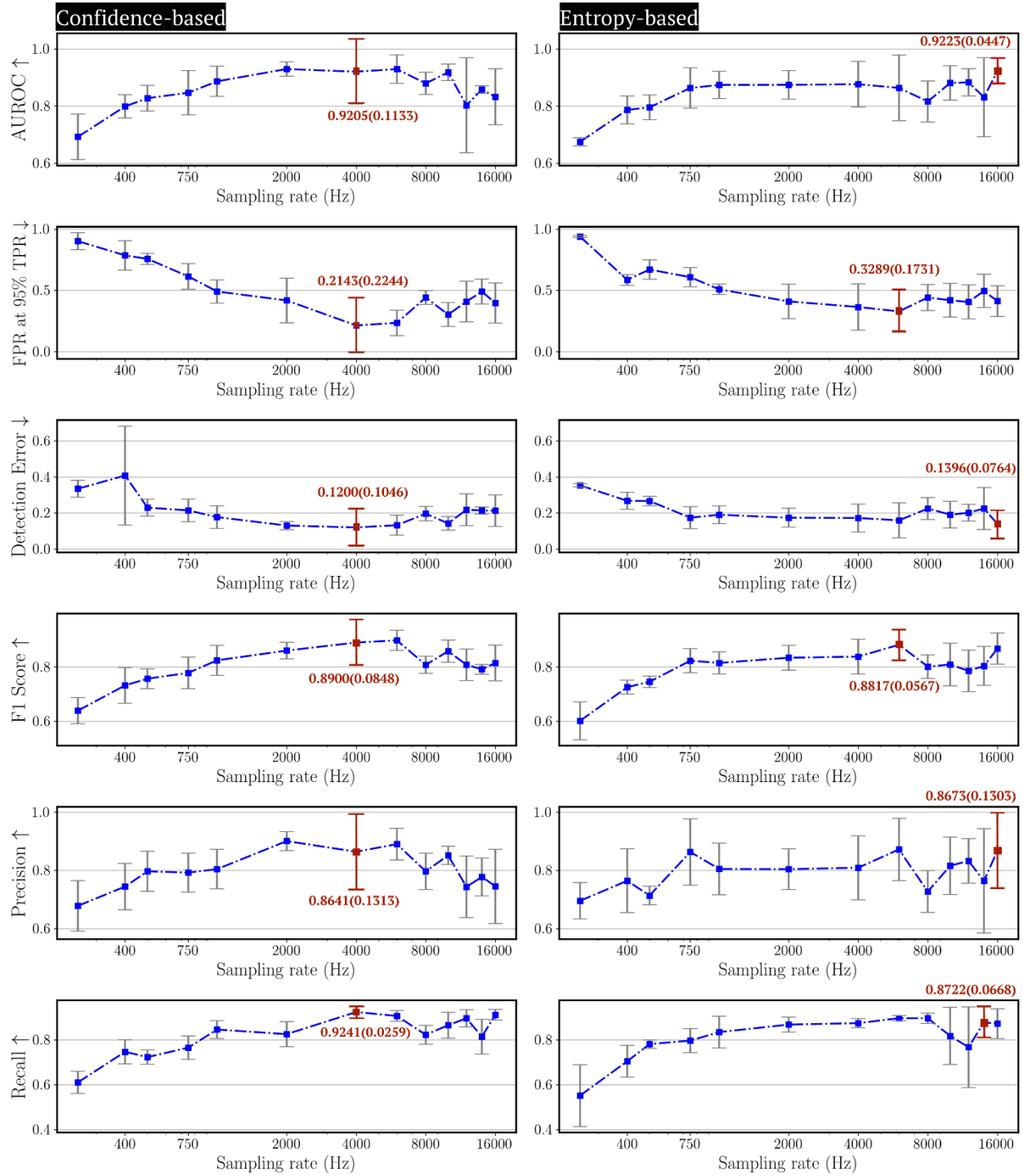


Figure 11. OOD detection task results comparison

In Fig. 11 we can see that the confidence-based model is able to achieve the highest AUROC, F1 score, and lowest FPR95 and detection errors for the 4kHz sampling frequency. Besides, all metrics keep relative promising values after 750 Hz sampling rate. It is also clear from the plots that at higher frequencies the variation increases and there is no generic upward or downward trends after 6kHz. For the entropy-based models, best performance is achieved at 16 kHz and results are less consistent after 4kHz. When compared with confidence-based models, the entropy-based models obtain better results at sampling rates of 750 Hz or higher.

For both models 750 Hz is the lowest sampling rate that gives acceptable results. At 4 kHz confidence-based model perform best and above 8 kHz the results become less consistent. Another thing to note here is that the AUROC value from the entropy-based model at 16 kHz is slightly higher than the confidence-based model at 4 kHz with lower standard deviation. The confidence-based model at 4 kHz performs better than the entropy-based model at 16 kHz but the entropy-based models are slightly better at dealing with higher frequencies.

4.2.4 Comparison of model performance on ID data mixed with OOD data

The last part of this phase focused on analyzing model's overall performance when OOD detection techniques discard OOD samples before a classification is made on ID data. We compare the OOD detection-based models with the baseline model and vary OOD proportions to find out at what percentage of OOD samples the baseline models start to degrade in performance. Note that the baseline models are based on the same architecture as OOD models (ResNet-50), but they do not have OOD detection techniques. When passing a mix of ID and OOD data we vary the proportion of OOD data from 0% to 100%, where at 0% there is no OOD data and at 100% the number of OOD data points equals the total number of ID data points, i.e., cough or speech data.

This does mean that at 100% the proportion of OOD data points is double the ID data points because cough and speech have equal number of data points in the ID test dataset.

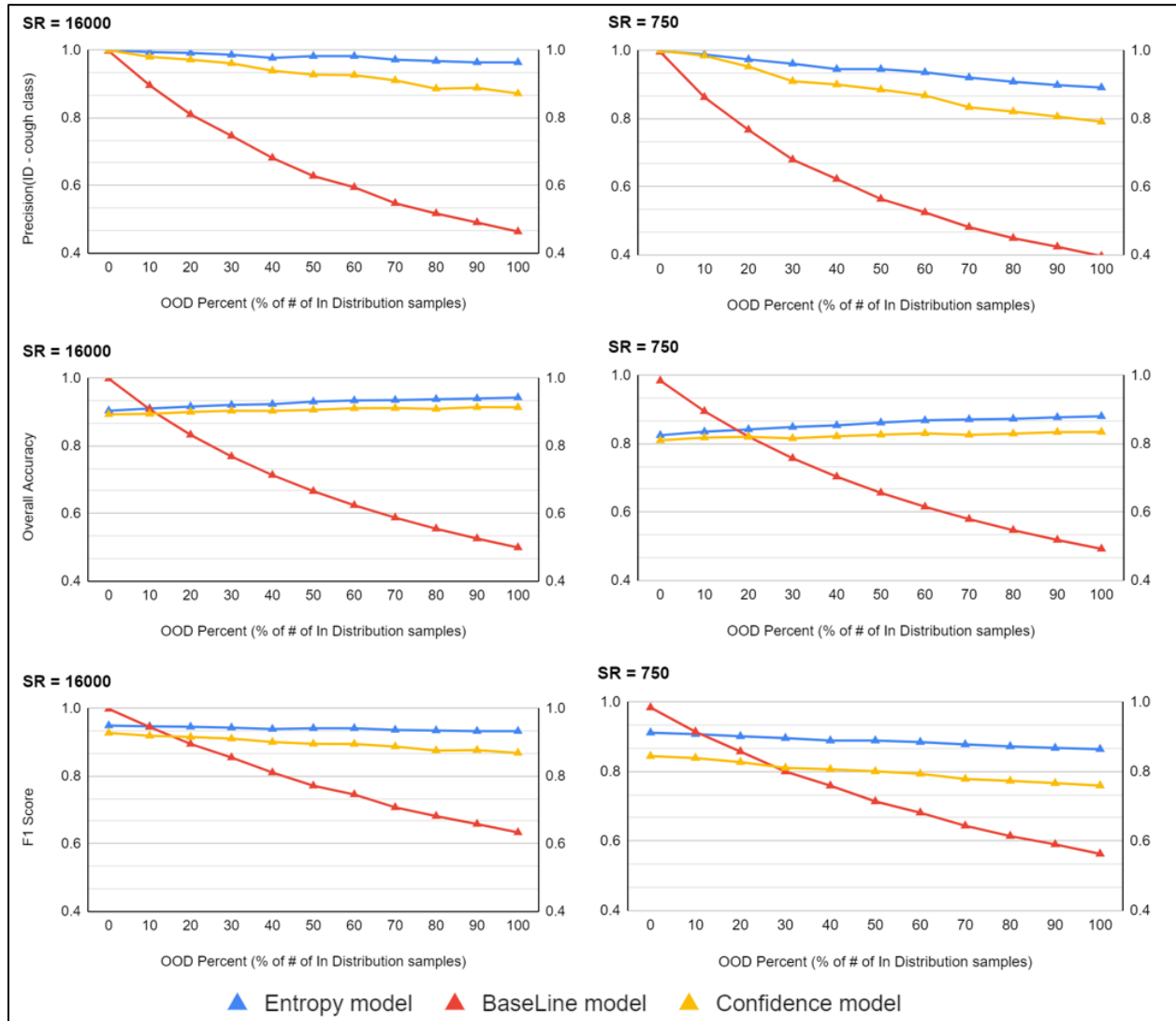


Figure 12. Model performance on ID data with varying OOD

Fig. 12 shows the comparison of performance of baseline and OOD detection models on ID data as OOD proportions are increased. The x-axis represents OOD data proportions that go from 0% to 100%, where at 0% there are no OOD data in the test dataset and at n% the total number of OOD samples equal n% of total number of ID samples. E.g., at 100% the OOD samples

equal ID-cough samples + ID-speech samples. Fig. 12 only shows a comparison for two sampling frequencies: 16KH and 750Hz and shows three different plots for Precision, F1 Score, and Overall accuracy. The Precision and F1 score plots show model's capacity in correctly recognizing ID-cough class samples among the ID-cough, ID-speech, and OOD class samples present in the test dataset. The Overall Accuracy plot on the other hand shows model's capacity in classifying samples as ID-cough, ID-speech, or OOD. In looking at accuracy plot the initial accuracy of baseline model is higher than that of OOD detection models but as OOD proportion increases the baseline model accuracy starts to degrade. This is because when there are no OOD data the baseline model correctly classifies samples as only ID-cough or ID-speech, but the OOD detection models incorrectly classifies some ID samples as OOD. But as OOD proportion increases the baseline model starts to make more mistakes in incorrectly classifying OOD samples as ID while the OOD detection models can discard OOD samples correctly before making classifications on ID samples. For the 16 kHz sampling rate the OOD models surpass the baseline model at an OOD proportion of around 10%, and the accuracy of the baseline drops from 99.89% to around 90%. For the 750 Hz dataset-based models the baseline performance becomes worse than OOD models at OOD proportion levels of about 15%, and the accuracy of the baseline drops from 98.44% to around 81%. The other thing to note on accuracy graphs is that the OOD models' accuracy increases with OOD proportions which happens because as OOD proportion is increased the OOD models are able to make more overall correct predictions, predictions that include correctly classifying OOD data with OOD class. For the F1 score and precision graphs as OOD data is introduced the performance of all models degrade but the baseline model performance degrades much faster than OOD models. When compared to confidence-based model, the entropy-based model has a lower

drop in performance on all metrics highlighting the fact that the entropy-based model seems to be equipped to deal with OOD data.

Because we are using the same ID data in all experiments the True Positive (TP) and False Negatives (FN) for the ID-cough class do not change and hence the recall scores do not change either with OOD proportions. Table 6 summarizes recall scores for all models.

Table 6. Recall scores comparison

Sampling Rate	Recall scores (%)		
	Baseline Model	Confidence OOD model	Entropy OOD model
16000 Hz	1	86.53	90.40
750 Hz	97.26	73.07	83.92

Overall, the recall scores for the baseline model are much better than OOD models but note that recall scores only account for TP and FN of ID-cough class samples and does not take into account False Positives and True Negatives. There are large gaps in performance at lower sampling rates than at higher sampling rates, e.g., for 750Hz sampling rate the recall score for baseline model is about 24% better than confidence based models and about 14% better than entropy based models. For 16 kHz, the difference is a little lower, with about 14% for confidence based model and only 10% for entropy based model.

4.3 Learnings

A sampling rate of 750Hz or above gives good results in terms of specificity, sensitivity, and F1 scores for correctly classifying cough in audio recordings and models with built-in OOD detection techniques perform better than models without OOD detection.

Chapter 5: Compare performance of time frequency functions

This chapter provides details of Phase III of the work, which concentrated on comparing performance of models trained on spectrograms generated using two different time-frequency analysis algorithms, Short-term Fast Fourier Transforms (St-FFT) vs SAS Short-term Parametric Power Spectral Density (St-PPSD). Sections below provide details on the datasets used and the results obtained on cough classification with OOD.

5.1 Datasets

We used the same datasets as described in Chapter 4.

5.2 Setup and Experiments

For these experiments we had to generate regular spectrograms based on St-FFT transformations and spectrograms based on St-PPSD transformations. As with Mel-spectrograms we used PyTorch torchaudio packages to create regular spectrograms and for St-PPSD we used SAS macros and SAS ESP servers where PPSD transformations can be generated. In addition to the ESP server, we also used low level functions in SAS to generate PPSD data. The downside with generating PPSD data is that it takes a lot of time when compared with FFT data using PyTorch packages. Due to this shortcoming, we were not able to test with all sampling frequencies and were forced to cut short our experiments to only use 400, 750, and 4kHz with 5 sec audio length samples. The other important point to note here is that unlike other experiments where we used Mel-Spectrograms in this experiment we only used regular spectrograms because at the time of conducting the experiments there were no available functionalities to convert PPSD datasets into mel scale.

For the neural network models, we used only the entropy based OOD detection model and did not use the baseline model or the confidence based OOD model. The experiments were conducted twice over the course of the study. The first time we used SAS Macros to generate PPSD datasets and used ResNet-50 architecture-based entropy OOD model. The process was slow because working with SAS Macros was very inefficient and took a lot of time. The second time the PPSD datasets were created we used SAS ESP servers, which was slow too compared to the time it takes to generate FFT transformations using PyTorch packages, but much faster than using SAS Macros. For the second dataset we changed our model architecture backbone from ResNet-50 to EfficientNetV2. We created 3 models for each sampling rates with both FFT and PPSD based datasets. Training was done for 6 epochs with a fixed learning rate of $1e-04$ and Adam optimizer. The best models were found by monitoring performance on the validation ID datasets. The performance metric that was tracked was the overall accuracy. Results reported in this study are obtained using the datasets created using the ESP server.

5.2.1 Why were the comparisons performed?

We performed these experiments because the quality of spectrogram images generated using FFT and PPSD transformations vary significantly. The PPSD transformation spectrograms have much less noise compared to the FFT spectrograms. Fig. 13 shows a comparison of the two spectrogram images for the same audio input. It is clear from the figure that PPSD spectrograms have less noise and cleaner features to learn as compared to FFT spectrogram. One would expect models trained on PPSD datasets to perform better than FFT spectrogram datasets.

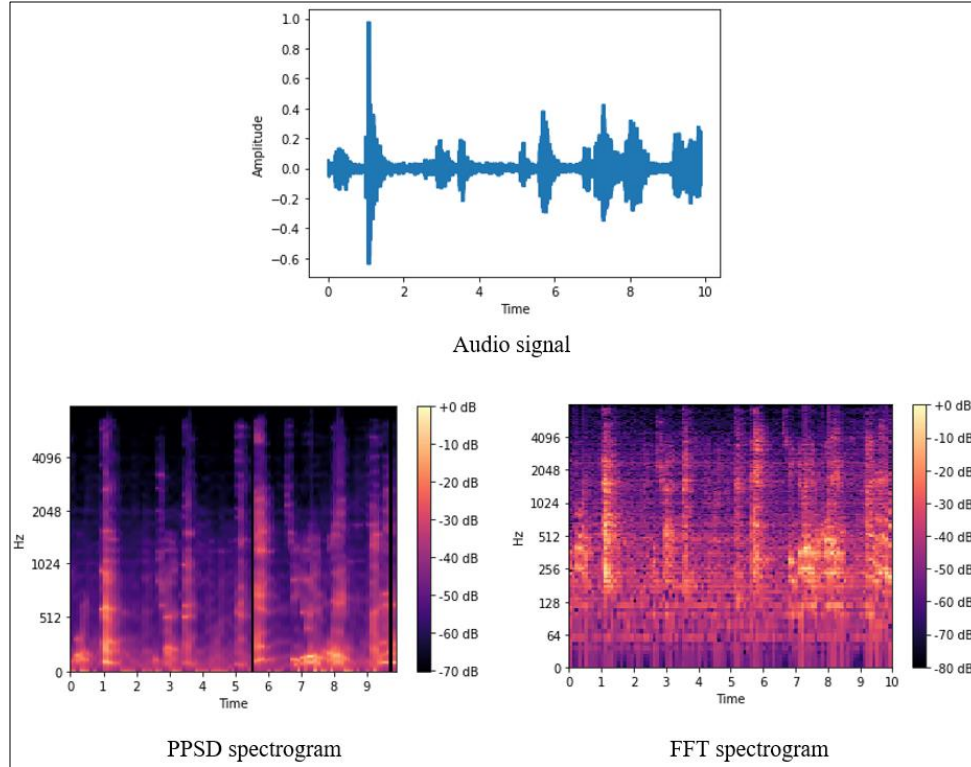


Figure 13. PPSD and FFT spectrogram

5.2.2 Performance comparison

The first experiment compares performance of models on the ID test dataset accuracy metrics. Accuracy refers to the percent of samples correctly classified as either ID-cough or ID-speech in the test dataset that had only cough and speech samples. Fig. 15 has results on the comparison. To our surprise models trained on PPSD datasets did not perform as well or better than the models trained on FFT datasets.

The Accuracy graph in Fig. 14 (left side) shows models with Regular Spectrograms, i.e., FFT spectrograms performs consistently better than PPSD dataset models for all tested sampling frequencies. The F1 score graph (right side), which shows models capacity in correctly distinguishing between ID and OOD samples shows a better performance for PPSD dataset models.

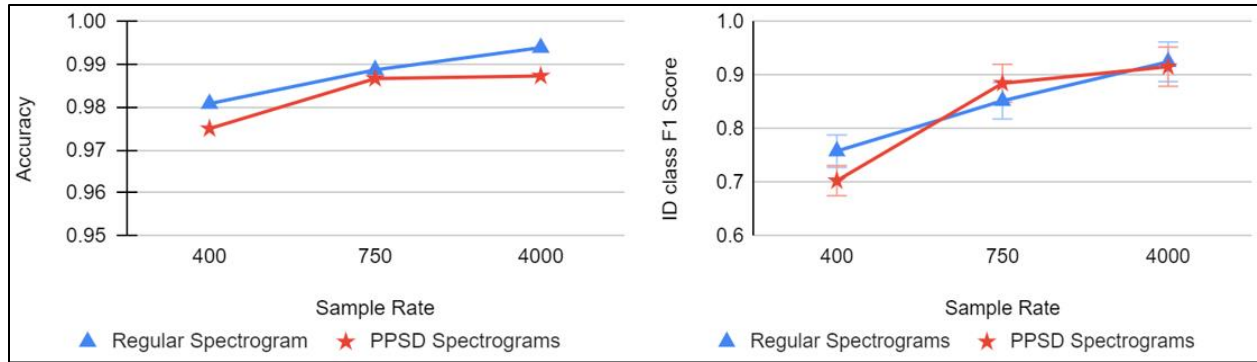


Figure 14. Performance comparison between PPSD and FFT dataset trained models

At 750Hz sampling frequency the PPSD datasets perform better than FFT and for 4K Hz sample rate both models perform about the same.

5.3 Learnings

PPSD dataset trained models did not exhibit any significant gains over FFT datasets. Note that the study was constrained because of the time it takes to generate PPSD datasets and hence future work is planned to test models at other sampling frequencies.

Chapter 6: Optimization of Mel Spectrograms for cough detection

This chapter provides details of the last phase, phase IV of the work. The data used in so far in our study consisted of 2D Mel Spectrogram images that were generated using a fixed set of spectrogram generation parameters across all frequencies. In this phase we conducted additional experiments to analyze the effect of changing the Mel Spectrogram generation parameters on model performance. We consider the model trained on datasets generated using the aforementioned fixed parameters as the baseline model to which we compare results of models trained on datasets created using other parameters. In this phase we only worked with 5 second audio samples at 750Hz but generated Mel Spectrogram images using 13 different spectrogram generation parameter combinations as listed in the left half of Table 6. The parameters tested were Window type, FFT size, # of Mel Filterbanks, Window Length, and Hop Length. As stated above we only experimented with the 5 second audio recordings, but for higher sampling rates the length of audio should be considered as another parameter to test.

6.1 Datasets

We used the same datasets as described in Chapter 4.

6.2 Setup and Experiments

In total around 20,000 samples of 5 seconds audios were generated from the datasets and were split into 75% training, 15% validation, and 10% test data split. When testing with OOD data we increase the proportion of OOD samples from 0 to 50% of ID data, where at 50% the number of OOD samples equal the number of cough or speech samples. Datasets were created for each of the 13 parameter combinations (highlighted in grey in Tables 7 and 8) and subsequently multiple

models were trained for each combination. For models we continued using EfficientNetV2 as our backbone and trained only the Entropy based OOD model. Mel spectrogram images generated for each combination were used to train and test 39 models (13 combinations x 3 models with different starting seeds). We set the learning rate to $1e-04$ and trained each model for 6 epochs. Each dataset had 14,000 ID samples in the training dataset, 2250 ID samples in the validation dataset, and 1604 ID samples in test dataset, equally split between cough and speech. OOD data is not used for training or validation and so we only created the test dataset for OOD that had 804 samples. For selecting the best model during training, validation dataset accuracy on ID dataset was monitored. Model that gave best accuracy on ID data, i.e., in detecting cough and speech samples, was chosen as the model for the combination. All scores reported in the tables below are mean/simple average of the scores obtained on the three models on test dataset. Note that the baseline model dataset was created using these parameters: Window Type: Hann Window, FFT Size: 1024, # of Mel Filterbanks: 128, Window Length: 1024, and Hop Length: 64.

6.2.1 Why were the comparisons performed?

Fig. 16 illustrates differences in images obtained using different parameters (a subset of the 13 combinations). The parameters are listed in the first column. The next three columns show Cough, OOD, and Speech audio Mel Spectrograms generated for each of the four combinations respectively. It is apparent from the images that the size and resolution of images are different for the models to learn different set of features and hence perform differently.

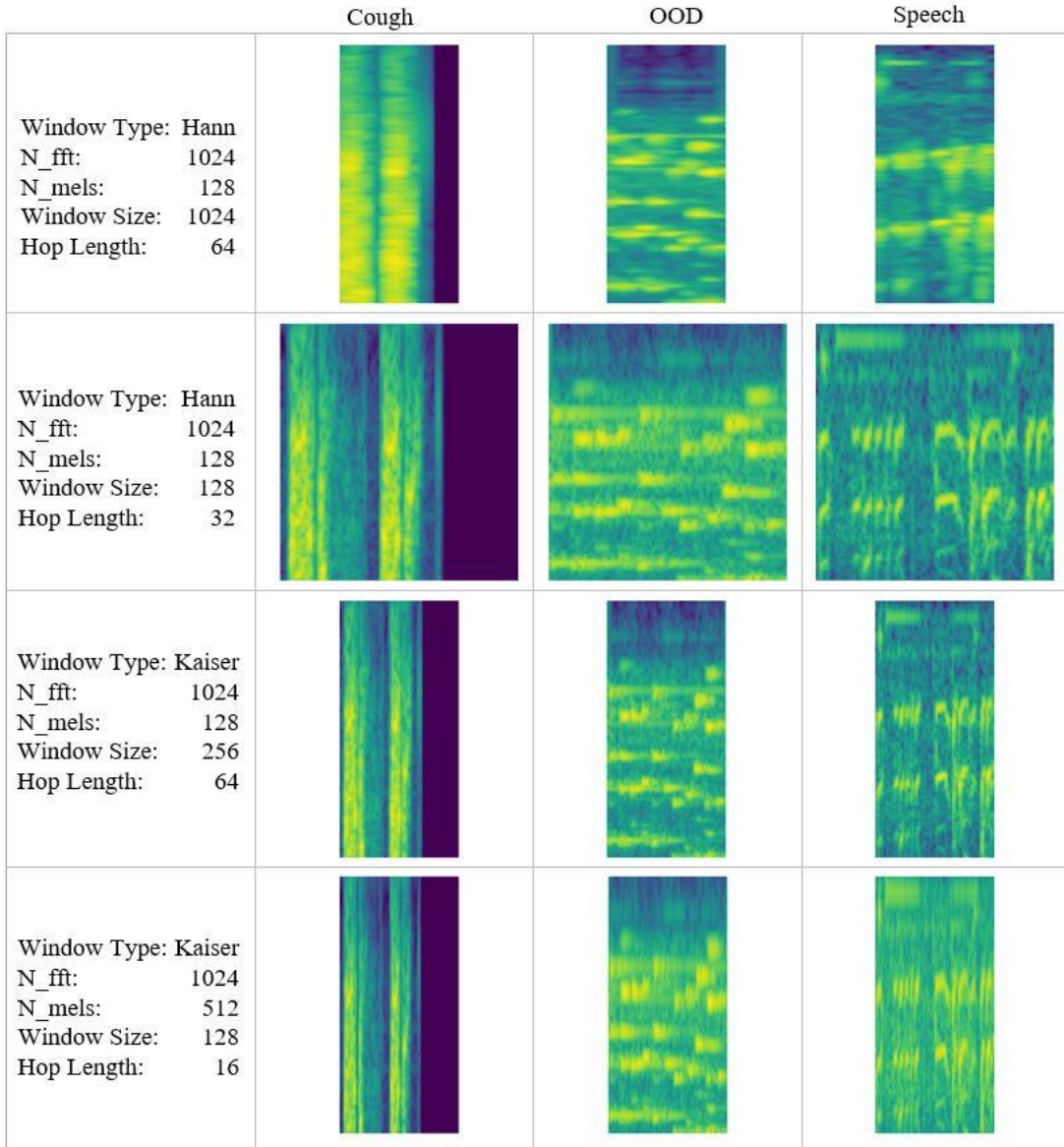


Figure 15. Mel spectrogram images for different parameter combinations

6.2.2 Model performance on ID data

Table 7 shows results of different model performances on ID data ID data correctly classified cough sample is classified as True Positive. Results highlight that models trained on datasets generated using kaiser windows generally perform better than the ones generated using Hann windows.

The model that gives best performance is based on the datasets generated using the following parameters: Window Type: Hann, FFT Size: 1024, # of Mel Filterbanks: 128, Window Length: 256, and Hop Length: 32, highlighted in bold color in Table 6. The baseline model results in the table are italicized and the second-best model results are shown in blue color. Compared to the baseline model the model with best parameter combination results gives an overall performance gain of about 1%.

Table 7. Model performance on In Distribution data for each combination of parameters.

Window Type	FFT Size	Mel Filters	Window Length	Hop Length	AUROC	Recall	Precision
Hann Window	512	128	256	32	98.88±0.39	98.88±0.39	98.79±0.50
	1024	128	128	32	99.00±0.65	99.00±0.65	98.88±0.37
	1024	128	256	32	99.44±0.16	99.44±0.16	99.29±0.19
	1024	128	256	64	98.42±0.42	98.43±0.42	98.30±1.10
	1024	128	512	32	99.11±0.53	99.11±0.53	98.67±0.68
	<i>1024</i>	<i>128</i>	<i>1024</i>	<i>64</i>	<i>98.52±0.37</i>	<i>98.53±0.36</i>	<i>98.50±0.66</i>
Kaiser Window	512	128	256	32	99.11±0.27	99.11±0.23	98.67±0.63
	1024	128	256	32	99.04±0.42	99.06±0.40	98.17±0.92
	1024	128	512	32	98.98±0.28	98.99±0.28	98.96±0.94
	1024	128	1024	64	98.38±0.53	98.39±0.53	97.92±1.06
	1024	256	256	32	99.36±0.16	99.36±0.16	99.04±0.07
	1024	512	128	16	99.13±0.28	99.14±0.27	98.63±0.77
	2048	128	1024	64	99.00±0.16	99.00±0.16	99.04±0.19

6.2.3 Model performance on OOD data

OOD detection performance is measured by the model's ability to distinguish between ID and OOD data. Table 8 reports results for various metrics for OOD performance. As with ID results the baseline model results are italicized, the parameter combination that gives best results are

highlighted in black bold, and second best performance model results are highlighted in blue. The combination with Window Type: Kaiser window, FFT size: 1024, # of Mel Filterbanks: 256, Window Length: 256, and Hop Length: 32, gives overall best results, although the best TNR @ 95% TPR is obtained for a Hann window combination highlighted in bold in Table 8. In comparison with the baseline model the model that gives best results has an overall performance gain of about 6-9% for the AUROC, Recall, Precision, and F1 score metrics.

Table 8. Model performance on OOD Data for each combination of parameters.

Window Type	FFT Size	Mel Filters	Window Length	Hop Length	AUROC	Recall	Precision	F1 Score	Detection Error	TNR at 95% TPR
Hann Window	512	128	256	32	90.16±3.27	83.79±3.47	84.21±2.94	83.72±3.55	13.97±4.46	59.10±11.4
	1024	128	128	32	93.12±3.65	88.35±6.04	88.60±5.82	88.32±6.07	11.64±6.04	59.78±15.7
	1024	128	256	32	91.93±4.07	85.13±6.68	85.42±6.73	85.10±6.69	14.86±6.68	61.72±14.7
	1024	128	256	64	90.21±3.01	84.59±3.66	85.24±3.92	84.52±3.66	15.41±3.66	42.31±12.7
	1024	128	512	32	92.86±3.15	88.10±1.76	88.51±1.31	88.07±1.81	11.89±1.76	49.66±23.8
	1024	128	1024	64	88.75±4.59	81.77±5.31	82.18±4.93	81.69±5.41	18.22±5.31	45.65±16.2
Kaiser Window	512	128	256	32	90.82±5.09	84.16±6.68	84.69±6.35	84.08±6.77	15.84±6.68	54.46±17.7
	1024	128	256	32	94.30±1.18	89.75±3.34	90.19±3.63	89.73±3.32	10.24±3.34	60.52±7.46
	1024	128	512	32	93.20±0.51	87.09±1.10	87.60±1.22	87.05±1.10	12.91±1.10	54.92±1.76
	1024	128	1024	64	90.97±0.53	85.12±0.55	85.92±0.56	85.04±0.55	14.87±0.55	40.77±10.6
	1024	256	256	32	94.47±0.24	90.08±1.56	90.37±1.68	90.06±1.56	9.92±1.57	60.05±9.10
	1024	512	128	16	92.89±3.28	87.85±5.98	88.53±5.55	87.78±6.06	12.15±5.98	54.53±8.09
	2048	128	1024	64	90.35±4.29	84.10±4.69	84.66±4.19	84.02±4.78	15.89±4.69	50.52±13.2

The OOD results have a lot more variability compared to ID results. This could be because OOD data is not used during validation and so a slight variation in finding Minimum Distance Score would affect ID performance marginally but can affect OOD performance significantly.

6.2.4 Model performance on In Distribution data with OOD detection

In this experiment we analyze performance of the models by first introducing OOD data in the test dataset and then letting the model discard OOD samples before making classifications. We increase the proportion of OOD data upto to 50%, where at 50% that there are as many OOD samples as there are cough or speech samples. Fig. 17 shows the performance of models created using different Mel spectrograms. In the figure we are only showing the top 5 best performing combinations and the worst performing combination in each graph for clarity. The baseline model performance is colored in black. The Overall Accuracy graph shows models ability to correctly classify samples as ID-cough, ID-speech, or OOD. In general, the Overall Accuracy of the models increase as the OOD percentage is increased but the baseline model accuracy reaches only 80% as compared to the best performing model with accuracy of about 89%. The Precision and F1 score graphs show an inverse trend with OOD proportions because as the OOD proportion increases the models make more mistakes in classifying OOD as ID data. At 50% OOD proportion levels the Precision and F1 scores for the best performing model are a respectable 93% and 90% compared to the baseline model's 85% and 83% scores. As with previous findings the Recall values do not change with increasing OOD proportions. As stated before, the results confirm that not all parameter combinations perform equally and that there is an optimal combination for a sampling frequency that outperforms others. In this experiment the models that gives the best ID

classification with OOD detection are obtained with Mel Spectrograms generated using the following parameters: Window Type: kaiser, FFT size: 1024, # of Mel Filterbanks: 256, Window Length: 256, and hop length: 32.

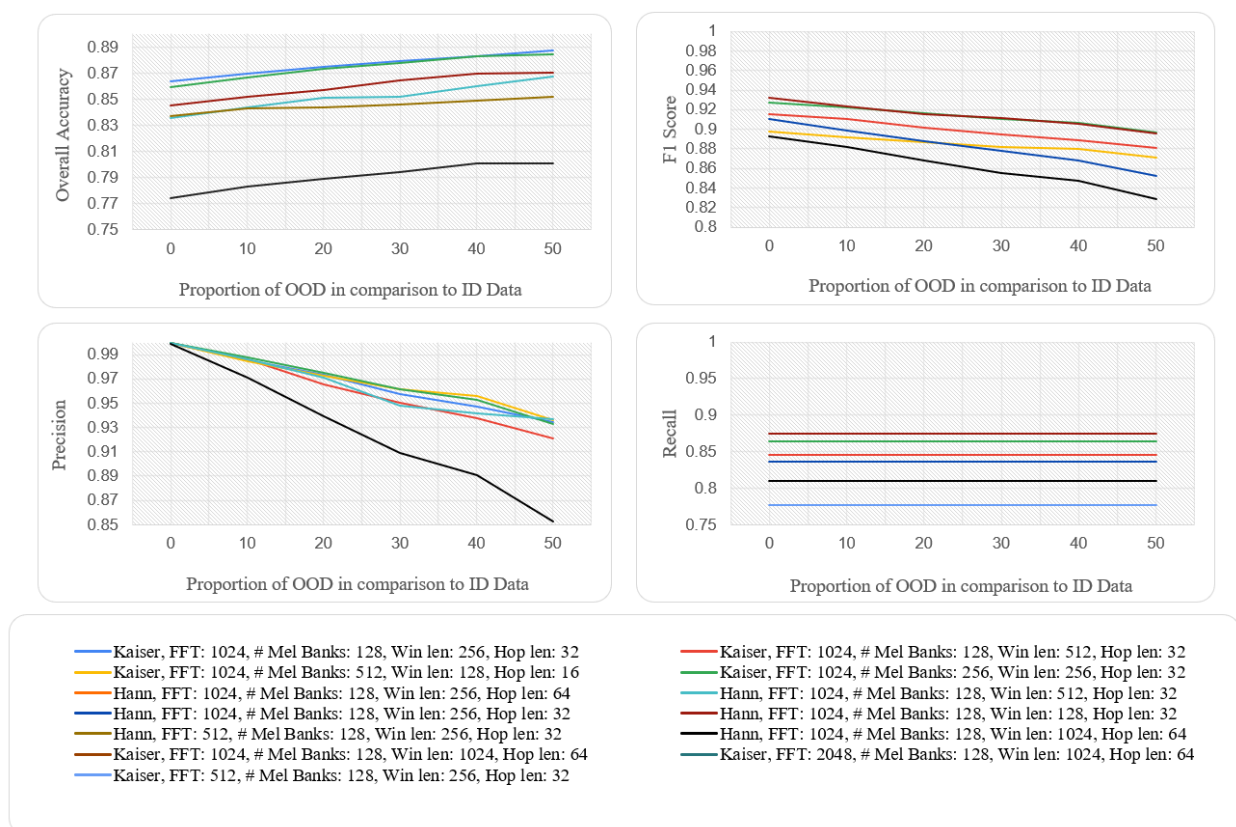


Figure 16. Model performance with varying OOD proportions

6.3 Learnings

The results of the experiment confirm the hypothesis that before training models we should investigate the hyper parameters to identify which combination will perform the best for a sample rate. This experiment was carried for only 750Hz sampling rate, but the results can be extrapolated to other sampling frequencies.

Chapter 7: Conclusions and Future work

We created a robust cough classification pipeline with OOD data detection techniques. The machine learning algorithms or models developed in this study can detect In-Distribution (ID) cough audio while detecting and discarding OOD audio. Implementation of OOD techniques do not significantly affect model performance on ID data. We performed a variety of experiments to find the minimum sampling frequency and length of audio recordings to be able to correctly classify cough and found that sampling rates greater than 750 Hz with audio lengths of 4-10 seconds can be used to convincingly detect cough. We also found some limitations of including OOD detection techniques in the models. If the audio recordings do not contain any OOD data and has only speech and cough signals, then models without OOD detection techniques perform much better than models with OOD techniques. On the other hand, if there is OOD data present the baseline models start to perform poorly as OOD proportions start increasing. We also used OOD data to find thresholds for both confidence based and entropy based OOD models to separate ID from OOD data. In real life we will not have access to OOD data and hence more work is planned to not have to use OOD data to find threshold values.

We also compared performance of models trained with Short-term Fast Fourier Transforms (St-FFT) based regular spectrogram datasets with datasets based on Short term Parametric Power Spectral Density (St-PPSD) and found that even though the PPSD datasets have less noise in the images the model performance did not translate to higher accuracy. The caveat here is that we performed limited number of experiments so we cannot be certain that this finding will hold true for all sampling frequencies. We plan to conduct more experiments with different sampling frequencies, model backbones, spectrogram generation parameters etc. to do a fair comparison.

We also proved that parameters used to generate 2D Mel Spectrogram images, i.e., datasets to train models have an impact on their performance. We tested 13 different combinations of parameters and found that some parameters perform better than others. We did not perform a systematic comparison of all viable possible combinations of parameters but instead chose a subset. We proved that when creating data for training neural network models an optimal combination of parameters should be sought. Experiments conducted in this study were restricted to 750Hz sampling frequency and a 5 sec audio interval, but the results can be easily extrapolated to other sampling frequencies and audio intervals. Our experiments showed that for models trained on the optimal combination of parameters there was a marginal gain of about 1% in ID data classification performance but a significant gain of about 8% in OOD detection performance as compared to models trained on the baseline combination of Mel Spectrogram generation parameters.

REFERENCES

- [1] Guo M Xie M Liu X Li X, Cao X. 2020. Trends and risk factors of mortality and disability adjusted life years for chronic respiratory diseases from 1990 to 2017: systematic analysis for the Global Burden of Disease Study 2017. *BMJ (Clinical research ed.)* (2020). <https://doi.org/10.1136/bmj.m234>
- [2] Foreman K et al. Lozano R, Naghavi M. 2012. Global and regional mortality from 235 causes of death for 20 age groups in 1990 and 2010: a systematic analysis for the Global Burden of Disease Study 2010. *Lancet* 380(9859) (2012), 2095–2128. [https://doi.org/10.1016/S0140-6736\(12\)61728-0](https://doi.org/10.1016/S0140-6736(12)61728-0)
- [3] C. Van Schayck and N. Chavannes, “Detection of asthma and chronic obstructive pulmonary disease in primary care,” *European respiratory journal*, vol. 21, no. 39 suppl, pp. 16s–22s, 2003.
- [4] Y. Chen, M. D. Wilkins, J. Barahona, A. J. Rosenbaum, M. Daniele, and E. Lobaton, “Toward automated analysis of fetal phonocardiograms: Comparing heartbeat detection from fetal doppler and digital stethoscope signals,” in 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). IEEE, 2021, pp. 975–978
- [5] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. Depristo, J. Dillon, and B. Lakshminarayanan, “Likelihood ratios for out-of-distribution detection,” in *Advances in Neural Information Processing Systems*, 2019, pp. 14 707–14 718.
- [6] A. Nguyen, J. Yosinski, and J. Clune, “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 427–436
- [7] Forsad Al Hossain, Andrew A Lover, George A Corey, Nicholas G Reich, and Tauhidur Rahman. 2020. FluSense: a contactless syndromic surveillance platform for influenza-like illness in hospital waiting areas. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–28.
- [8] Justice Amoh and Kofi Odame. 2016. Deep Neural Networks for Identifying Cough Sounds. *IEEE Transactions on Biomedical Circuits and Systems* 10, 5 (2016), 1003–1011. <https://doi.org/10.1109/TBCAS.2016.2598794>
- [9] K. J. Piczak, “Esc: Dataset for environmental sound classification,” in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1015–1018.

- [10] E. Fonseca, M. Plakal, F. Font, D. P. Ellis, X. Favory, J. Pons, and X. Serra, “General-purpose tagging of freesound audio with audioset labels: Task description, dataset, and baseline,” arXiv preprint arXiv:1807.09902, 2018.
- [11] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2015, pp. 5206–5210
- [12] M. Abdelkhalek, J. Qiu, M. Hernandez, A. Bozkurt, and E. Lobaton, “Investigating the relationship between cough detection and sampling frequency for wearable devices,” in 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). IEEE, 2021, pp. 7103–7107
- [13] M. Al-Khassaweneh and R. Bani Abdelrahman, “A signal processing approach for the diagnosis of asthma from cough sounds,” *Journal of medical engineering & technology*, vol. 37, no. 3, pp. 165–171, 2013
- [14] S. Matos, S. S. Birring, I. D. Pavord, and H. Evans, “Detection of cough signals in continuous audio recordings using hidden markov models,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 1078–1083, 2006
- [15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In 2009 IEEE Conference on Computer Vision and Pattern Recognition. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
- [16] Terrance DeVries and Graham W Taylor. 2018. Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018).
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770–778.
- [18] Anurag Kumar, Maksim Khadkevich, and Christian Fügen. 2018. Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 326–330.

- [19] David Macêdo, Tsang Ing Ren, Cleber Zanchettin, Adriano LI Oliveira, and Teresa Ludermir. 2021. Entropic Out-of-Distribution Detection: Seamless Detection of Unknown Examples. *IEEE Transactions on Neural Networks and Learning Systems* (2021).
- [20] Igor DS Miranda, Andreas H Diacon, and Thomas R Niesler. 2019. A comparative study of features for acoustic cough detection using deep architectures. In 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2601–2605.
- [21] R. X. A. Pramono, S. A. Imtiaz, and E. Rodriguez-Villegas, “Automatic cough detection in acoustic signal using spectral features,” in 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2019, pp. 7153–7156
- [22] J. Monge-Álvarez, C. Hoyos-Barceló, L. M. San-José-Revuelta, and P. Casaseca-de-la Higuera, “A machine hearing system for robust cough detection based on a high-level representation of band-specific audio features,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 8, pp. 2319–2330, 2018
- [23] G.-T. Lee, H. Nam, S.-H. Kim, S.-M. Choi, Y. Kim, and Y.-H. Park, “Deep learning based cough detection camera using enhanced features,” *Expert Systems with Applications*, p. 117811, 2022
- [24] S. Jokić, D. Cleres, F. Rassouli, C. Steurer-Stey, M. A. Puhan, M. Brutsche, E. Fleisch, and F. Barata, “Tripletcough: Cougher identification and verification from contact-free smartphone-based audio recordings using metric learning,” *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 6, pp. 2746–2757, 2022.
- [25] D. Hendrycks and K. Gimpel, “A baseline for detecting misclassified and out-of-distribution examples in neural networks,” *arXiv preprint arXiv:1610.02136*, 2016.
- [26] S. Liang, Y. Li, and R. Srikant, “Enhancing the reliability of out-of-distribution image detection in neural networks,” *arXiv preprint arXiv:1706.02690*, 2017.
- [27] T. DeVries and G. W. Taylor, “Learning confidence for out-of-distribution detection in neural networks,” *arXiv preprint arXiv:1802.04865*, 2018.

- [28] G. Shalev, Y. Adi, and J. Keshet, “Out-of-distribution detection using multiple semantic label representations,” in *Advances in Neural Information Processing Systems*, 2018, pp. 7375–7385.
- [29] A. Subramanya, S. Srinivas, and R. V. Babu, “Confidence estimation in deep neural networks via density modelling,” arXiv preprint arXiv:1707.07013, 2017.
- [30] T. Denouden, R. Salay, K. Czarnecki, V. Abdelzad, B. Phan, and S. Vernekar, “Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance,” arXiv preprint arXiv:1812.02765, 2018.
- [31] V. Abdelzad, K. Czarnecki, R. Salay, T. Denouden, S. Vernekar, and B. Phan, “Detecting out-of-distribution inputs in deep neural networks using an early-layer output,” arXiv preprint arXiv:1910.10307, 2019.
- [32] K. Lee, H. Lee, K. Lee, and J. Shin, “Training confidence-calibrated classifiers for detecting out-of-distribution samples,” arXiv preprint arXiv:1711.09325, 2017.
- [33] D. Hendrycks, M. Mazeika, and T. Dietterich, “Deep anomaly detection with outlier exposure,” arXiv preprint arXiv:1812.04606, 2018.
- [34] S. Vernekar, A. Gaurav, V. Abdelzad, T. Denouden, R. Salay, and K. Czarnecki, “Out-of-distribution detection in classifiers via generation,” arXiv preprint arXiv:1910.04241, 2019.
- [35] Lara Orlandic, Tomas Teijeiro, and David Atienza. 2021. The COUGHVID crowd-sourcing dataset, a corpus for the study of large-scale cough analysis algorithms. *Scientific Data* 8, 1 (2021), 1–10.
- [36] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. <https://doi.org/10.48550/ARXIV.1409.1556>
- [37] David Snyder, Guoguo Chen, and Daniel Povey. 2015. Musan: A music, speech, and noise corpus. arXiv preprint arXiv:1510.08484 (2015).
- [38] Mingxing Tan and Quoc V. Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. (2019). <https://doi.org/10.48550/ARXIV.1905.11946>
- [39] Mingxing Tan and Quoc V. Le. 2021. EfficientNetV2: Smaller Models and Faster Training. (2021). <https://doi.org/10.48550/ARXIV.2104.0029>

- [40] M. Huzaifah, "Comparison of time-frequency representations for environmental sound classification using convolutional neural networks," arXiv preprint arXiv:1706.07156, 2017